

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**  
**СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ**

Кафедра прикладної математики та моделювання складних систем

Допущено до захисту

Завідувач кафедри ПМ та МСС

\_\_\_\_\_ Коплик І.В

(підпис)

«\_\_» \_\_\_\_\_ 20\_\_ р.

**КВАЛІФІКАЦІЙНА РОБОТА**

на здобуття освітнього ступеня «магістр»

спеціальність 113 «Прикладна математика»

освітньо-професійна програма «Прикладна математика»

тема роботи: **«СТАБІЛІЗАЦІЯ ПОВЕРХНЕВИХ СТРУКТУР АДСОРБАТУ  
ПРИ КОНДЕНСАЦІЇ ЗА РАХУНОК ДИФУЗІЇ АДАТОМІВ МІЖ  
ШАРАМИ»**

**Виконавець**

студент факультету ЕЛІТ

Глущенко Богдан Алімович \_\_\_\_\_

(підпис)

**Науковий керівник**

кандидат фізико-математичних наук

Дворниченко Аліна Василівна \_\_\_\_\_

(підпис)

Суми 2020

## СУМСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ

Факультет Кафедра	<b>електроніки та інформаційних технологій прикладної математики та моделювання складних систем</b>
Рівень вищої освіти	<u>другий (магістр)</u>
Галузь знань	<b>11 Математика та статистика</b>
Спеціальність	<b>113 Прикладна математика</b>
Освітня програма	<b>освітньо-професійна «Наука про дані та моделювання складних систем»</b>

ЗАТВЕРДЖУЮ

Завідувач кафедри ПМтаМСС

Коплик І.В. \_\_\_\_\_

«\_\_» \_\_\_\_\_ 20\_\_ р.

### З А В Д А Н Н Я

#### НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧУ ВИЩОЇ ОСВІТИ

Глущенко Богдану Алімовичу

1. Тема роботи Стабілізація поверхневих структур адсорбату при конденсації за рахунок дифузії адатомів між шарами

Керівник роботи Кандидат фізико-математичних наук, Дворниченко Аліна Василівна

затверджую наказом по факультету ЕлІТ від «12» жовтня 2020 р. № 1548-III

2. Термін подання роботи студентом «16» грудня 2020р.

3. Вихідні данні до роботи 1) Реакційно-дифузійна модель конденсації з газової фази;

2) Дифузійний потік.

4. Зміст розрахунково-пояснювальної записки (перелік питань, що їх належить розробити) Встановлення залежності стаціонарної концентрації адсорбату від коефіцієнту дифузії адатомів між шарами в однорідній системі, встановлення стійкості стаціонарних станів шляхом аналізу показника Ляпунова та відповідного потенціал, проведення моделювання динаміки

еволюції концентрації адсорбату в однорідній системі з використанням різних числових методів, встановлення умови реалізації стаціонарних просторових структур шляхом аналізу стійкості однорідних стаціонарних станів до неоднорідних збурень.

---

5. Перелік графічного матеріалу

1) Залежність концентрації адсорбату від коефіцієнта дифузії.

2) Залежність  $\lambda_0$  від  $k$ .

3) Фазова діаграма залежності вертикальної дифузії що залежить від матеріалу.

4) Залежність потенціалу в різних областях діаграми.

5) Поведінка еволюції системи при різних стаціонарних точках (одна точка).

6) Поведінка еволюції системи при різних стаціонарних точках (три точки).

7) Залежність показника Ляпунова.

8) Діаграма залежності.

9) Просторові структури що відповідають моделюванню.

3. Консультанти до проекту (роботи), із значенням розділів проекту, що стосується їх

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання «02» листопада 2020р.

### КАЛЕНДАРНИЙ ПЛАН

№ п/п	Назва етапів кваліфікаційної роботи	Термін виконання роботи	Примітка
1)	Ознайомлення з поставленою задачею	02.11.2020- 07.11.2020	
2)	Пошук і аналіз літератури	08.11.2020- 11.11.2020	
3)	Проведення розрахунків	11.11.2020- 18.11.2020	
4)	Моделювання та опис отриманих розрахунків	18.11.2020- 30.11.2020	

Здобувач вищої освіти

\_\_\_\_\_

(підпис)

\_\_\_\_\_

(прізвище та ініціали)

Керівник роботи

\_\_\_\_\_

(підпис)

\_\_\_\_\_

(прізвище та ініціали)

## РЕФЕРАТ

**Кваліфікаційна робота:** 56 с., 9 рисунків, 14 джерел.

**Мета роботи:** визначити вплив коефіцієнту дифузії адатомів між шарами на конфігурацію системи газ-конденсат у стаціонарному режимі.

**Об'єкт дослідження:** реакційн-дифузійна модель конденсації з газової фази.

**Предмет дослідження:** модель формування просторових структур в області нестабільності системи.

**Методи навчання:** реакційно-дифузійна модель, потенціал, метод Єйлера та Рунге-Кута, Перетворення Фур'є, показник Ляпунова, оператор Свіфта-Хогенбера, сферична система координат.

Для дослідження системи на стійкість знайдено залежності коефіцієнту дифузії до концентрації адсорбату. Для демонстрації поведінки області в якій існує перехід першого роду будуємо потенціали. Для дослідження поведінки еволюції системи при різних стаціонарних точках було досліджено з використанням методу Єйлера та методу Рунге-Кута. Досліджуємо зміну морфології поверхні та динаміку утворення поверхневих структур шляхом числових симуляцій.

**Ключові слова:** ДЕСОРБЦІЯ, АДСОРБЦІЯ, СТІЙКИЙ ТА НЕСТІЙКИЙ СТАН, ДИФУЗІЙНИЙ ПОТІК, ПОВЕРХНЕВІ СТРУКТУРИ, ОДНОРІДНА СИСТЕМА.

## ЗМІСТ

Вступ.....	7
Літературний огляд.....	9
Постанова задачі.....	11
1. Моделювання.....	12
1.1 Реакційно-дифузійна модель конденсації з газової фази.....	12
2.1 Дифузійний потік.....	14
3.1 Стаціонарні стани однорідної системи та їх стійкість.....	16
4.1 Умови реалізації стійких просторових структур.....	21
5.1 Моделювання росту поверхневих структур.....	23
Висновок.....	26
Список літератури.....	27
Додаток А.....	29
Додаток В.....	39
Додаток С.....	49

## ВСТУП

В останні кілька десятиліть нанофізика стає все більш цікавою для сучасної науки. В першу чергу це пов'язано з виробництвом сучасних інструментів, електронної техніки та швидким розвитком суспільства. Наноструктуровані об'єкти в сучасній наноелектроніці, пристроях і пристроях зв'язку, біомедицині та ін. Підвищена увага до нанорозмірних об'єктів є результатом їх унікальних властивостей і функціональності.

До останніх відносяться гігантські магніти, контрольоване оптичне випромінювання, висока ефективність фотоелектричного перетворення, наднизька теплопровідність тощо. Ці унікальні особливості дозволяють використовувати такі матеріали в магніторезистивних датчиках, пристроях пам'яті, оптичних і комунікаційних пристроях, лазерах і детекторах на квантових точках.

У тонкоплівкових системах часто спостерігається утворення наноструктурованих об'єктів, що робить їх ще більш важливими. Тому актуальним питанням в даний час є з'ясування ролі основних механізмів, які призводять до створення наноструктурованих тел, особливо в тонких плівках, і встановлення механізмів управління типом і розміром стаціонарних наноструктур на поверхні тонких плівок.

Результати теоретичних і експериментальних досліджень показують, що використання різних матеріалів (металів, напівпровідників) дозволяє вирощувати наноструктури різного типу і розміру, тобто наноточек, наностержней і подовжених структур.

Нанотехнології займають особливе місце серед сучасних сфер людської діяльності. Слід зазначити, що ідея створення пристрою з найменшими габаритами була недалеко від нас. Теоретичні та експериментальні спостереження показують, що реакційно-дифузійні системи відіграють важливу роль у вивченні загального просторово-часового поведінки незбалансованих систем.

Експериментальні методи, такі як польова іонна мікроскопія і скануюча тунельна мікроскопія, можуть використовуватися для моніторингу хімічних реакцій на поверхні металу на атомному рівні. У процесах адсорбції-десорбції, коли матеріал може розкладатися з газової фази, ці експериментальні методи дозволяють вивчати утворення кластерів або острівців адсорбованих молекул/атомів, які можуть мати лінійні розміри в нанометровому діапазоні. Експериментально підтверджено появу наноструктур на твердій поверхні і нанополе в адсорбованих моноатомній шарах.

На сьогоднішній день існує ряд методів вирощування тонких наноструктурованих плівок. Вони можуть включати: хімічне осадження з парової фази (осадження при атмосферному тиску, осадження при низькому тиску або осадження в високому вакуумі), фізичне осадження (з використанням електронного променя або імпульсного лазера), гомо- і гетероепітаксис.

Використовуючи різні методи осадження, можна ефективно застосовувати механізми зростання нанорозмірних острівців адсорбатов: адсорбція-десорбція, дифузія, атомне взаємодія. В останні десятиліття заклепки і подовжені нанострікційні структури спостерігалися в реальних експериментах і в процесах осадження з парової фази з використанням чисельного моделювання та електронних пучків.



## ЛІТЕРАТУРНИЙ ОГЛЯД

Відомо, що короткочасні переходи можна спостерігати на ранніх стадіях фазового розшарування [1]. Такі тимчасові моделі можуть бути стабілізовані дисбалансом хімічних реакцій, які призводять до появи стаціонарних моделей.

Адсорбція та термодесорбції є «рівноважними реакціями» і тому не можуть призводити до модульованої фазової стабільності з точки зору кінетичного простору. В адсорбційних і десорбційних системах необхідно застосовувати додаткові хімічні реакції для заморожування моделей.[2]

Проблемі освіти стаціонарних мікроструктур в таких системах в результаті незворотних нерівноважних хімічних реакцій присвячені наукові статті.[3] Дослідження показали, що стохастичний внесок залежить від комбінацій дисипації і пульсації в чистих дисипативних системах, які можуть підтримувати формування стабільних структур в нанорозмірних діапазонах.[4]

Зазвичай ці моделі роблять значний внесок як в локальну динаміку (хімічні реакції, такі як народження-смерть), так і в масоперенос.

Передові експериментальні методи, такі як польова іонна мікроскопія і скануюча тунельна мікроскопія, використовуються для спостереження хімічних реакцій на металевих поверхнях на атомистическом рівні. [5] Таким чином, коли матеріал може осідати з газової фази в процесах адсорбції-десорбції, такі експериментальні методи дозволяють вивчати освіту груп або острівців лінійних наномасштабів адсорбованих молекул / атомів.[6]

Такі структури в масштабах, менших, ніж довжина дифузії, виникають в результаті квазіхімічних реакцій і взаємодій між адсорбованими частинками. Присутність адсорбатів може вплинути на локальні кристалографічні структури в поверхневому шарі підкладки і викликати великі взаємодії між адсорбованими атомами і їх групами.[7]

Сьогодні використання тонких плівок є найбільш поширеним методом поліпшення фізичних і хімічних властивостей матеріалів практично у всіх

областях матеріалознавства. [8] Тонка плівка використовується в багатьох інженерних системах і широко використовується для різних функцій. Значний і значний прогрес, особливо в технології тонких плівок, сприяв швидкому розвитку мініатюризації електронних пристроїв.[9]

Використання електричного заряду в таких пристроях залежить від кордону між матеріалами з різними електронними властивостями. Крім того, потреба в високоякісних, високоякісних і надійних тонкоплавких матеріалах стала рушійною силою швидкого розвитку і значного прогресу в технології збільшення тонких плівок.[10] Значний прогрес у фізиці тонких наноструктурованих плівок ставить нові завдання для їх синтезу та моделювання. Інший приклад використання тонких плівок - їх інтеграція в мікроелектромеханічні системи. [11]

У минулому столітті тонкі плівки залучили інтерес вчених і інженерів і стали предметом інтенсивних досліджень і розробок. Зокрема, випробування проводилися на тонких плівках на відносно товстих підлогах, багат шарових матеріалах і конструкційних плівках на підлозі. Рушійними силами сучасних технологічних процесів стали суттєві досягнення в методах синтезу, обробки та визначення характеристик цих матеріалів. [12]

Дослідження показують, що тонкі розплавлені матеріали стали важливою частиною досягнень в технології матеріалів у другій половині 20-го століття: електроніка, комп'ютери, медичні технології, лазери і оптичні волокна. [13] Розвиток і просування багатьох методів і підходів до вирощування тонких плівок і управління ними призвело до їх широкому використанню в промисловості. Рання індустріалізація технології тонкого палива була ключовим фактором подальшого розвитку і вдосконалення методів осадження.[14]

## ПОСТАНОВА ЗАДАЧІ

**Мета:** визначити вплив коефіцієнту дифузії адатомів між шарами на конфігурацію системи газ-конденсат у стаціонарному режимі.

1. Встановити залежності стаціонарної концентрації адсорбату від коефіцієнту дифузії адатомів між шарами в однорідній системі.

2. Встановити стійкість стаціонарних станів шляхом аналізу показника Ляпунова та відповідного потенціалу.

3. Провести моделювання динаміки еволюції концентрації адсорбату в однорідній системі з використанням різних числових методів.

4. Встановити умови реалізації стаціонарних просторових структур шляхом аналізу стійкості однорідних стаціонарних станів до неоднорідних збурень.

# 1. МОДЕЛЮВАННЯ

## 1.1 РЕАКЦІЙНО-ДИФУЗІЙНА МОДЕЛЬ КОНДЕНСАЦІЇ З ГАЗОВОЇ ФАЗИ

Для вивчення поведінки наночасток адсорбата при конденсації з газової фази ми розглядаємо модель тільки з одним типом частинок. Припускаючи, що всі попередні шари повністю заповнені адсорбатом, будемо розглядати одношарову модель. При вивченні процесів конденсації з газової фази на мезоскопічеські рівні зручно розділити всі поверхневі осередки і ввести локальну концентрацію адсорбатов в осередку, як відношення кількості адсорбованих частинок у комірці поверхні до фіксованої кількості доступних вузлів у кожній комірці  $x(\mathbf{r}, t) \in [0; 1]$ ;  $t$  – часова змінна,  $\mathbf{r}$  – просторова координата. Таким чином, при описі процесу конденсації з газової фази еволюція системи задається реакційно-дифузійним рівнянням на поле концентрації адсорбату у стандартному вигляді:

$$\frac{\partial x}{\partial t} = f(x) + \nabla^2 J, \quad (1.1)$$

де  $f(x)$  – відповідає за квазіхімічній реакції, визначає процеси адсорбції/десорбції та позначається:

$$f(x) = f_{ads} + f_{des} + f_{vd} \quad (1.2)$$

Адсорбція визначається як:

$$f_{ads} = (1 - x)(1 - y) \quad (1.3)$$

Десорбція визначається як:

$$f_{des} = -x(1 - y)e^{-2\epsilon x} \quad (1.4)$$

Вертикальна дифузія між шарами визначається як:

$$f_{vd} = -kx(1 - y) + ky(1 - x) = -kx + kxy + ky - kxy = k(x - y) \quad (1.5)$$

Адсорбція і десорбція - це рівноважні реакції, які призводять до обміну частками між покриттям і, особливо, резервуаром. У замкнутій системі цього типу можливі тільки просторові обурення за короткий проміжок часу, що призводить до застосування перехідних структур. В іншому випадку система прагне до термодинамічної рівноваги, що означає, що підкладка покрита адсорбатом. Щоб продемонструвати формування стаціонарних структур і

стабілізацію процесу освіти ізольованих структур, необхідно провести узагальнення по моделі з урахуванням незворотних реакцій в дисбалансі, пов'язаних з утворенням стійких комплексів.

Еволюція структур субстрату змінюється при наявності незбалансованих хімічних реакцій. Припускаючи, що адсорбція можлива тільки за відсутності адсорбатів ( $x=1$ ) після всіх цих реакцій (з урахуванням нерівноважних реакцій), компонент реакції рівняння виглядає наступним чином:

$$f(x) = \alpha(1-x)(1-y) - x(1-y)e^{-2\epsilon x} + k(y-x) \quad (1.6)$$

де  $x$  – концентрація адсорбату,  $\nabla^2 J$  – просторова похідна, визначає перенесення. Припускаємо що  $y = \gamma x$ ,  $\gamma < 1$ ,  $\gamma$  – коефіцієнт, який залежить від матеріалу.

## 2.1 ДИФУЗІЙНИЙ ПОТІК

Оскільки адатоми є рухливими взаємодійними частинками, то повний потік адсорбату  $J$  задається сумою звичайного дифузійного потоку  $(-D\nabla x)$  і потоку адсорбату  $(-\left(\frac{D}{T}\right)x(1-x)\nabla U(\mathbf{r}))$ , визначеного потенціалом взаємодії. Тут множник  $x(1-x)$  вказує на те що, потік можливий лише на  $(1-x)$  вільних вузлів. Тоді, уводячи мобільність  $M(x) = x(1-x)$ , повний потік можна переписати у вигляді:

$$J = -DM(x)\left[\frac{\nabla x}{x(1-x)} - \frac{1}{T}\nabla U\right] \quad (2.1)$$

Формально, цей потік можна записати у канонічному вигляді:

$$J = -DM(x)\nabla\frac{\delta F}{\delta x} \quad (2.2)$$

де повний функціонал вільної енергії визначається сумою невзаємодійної  $F_0$  та взаємодійної  $F_{int}$  частин. Невзаємодійна частина має форму

$$F_0 = \int dr [x(\mathbf{r})\ln(x(\mathbf{r})) + (1-x(\mathbf{r}))\ln(1-x(\mathbf{r}))] \quad (2.3)$$

у той час як  $F_{int}$  визначається потенціалом взаємодії  $U(\mathbf{r})$  стандартним чином:  $F_{int} = 1/2 \int x(\mathbf{r})U(\mathbf{r}-\mathbf{r}')d\mathbf{r}$ . Потенціал взаємодії  $U(\mathbf{r})$  напишемо у наступному вигляді:

$$U(\mathbf{r}) = - \int d\mathbf{r}' U(\mathbf{r}-\mathbf{r}')x(\mathbf{r}') \quad (2.4)$$

Припускаючи, що концентрація адсорбату  $x$  слабо змінюється в межах радіуса взаємодії, можна використати апроксимацію.

Підставляючи і враховуючи члени до 4-го порядку, одержуємо

$$\int u(r)x(r)dr = 2\varepsilon x, \quad (2.5)$$

$$\frac{1}{2!}(\int u(r)r^2 dr)\nabla^2 x(r) = 2\varepsilon r_0^2 \nabla^2 x, \quad (2.6)$$

$$\frac{1}{4!}(\int u(r)r^4 dr)\nabla^4 x(r) = \varepsilon r_0^4 \nabla^4 x \quad (2.7)$$

Таким чином, використовуючи позначення  $\varepsilon = \varepsilon/T$  одержуємо:

$$\frac{1}{T} \int d\mathbf{r}' U(\mathbf{r}-\mathbf{r}')x(\mathbf{r}') = \varepsilon x(\mathbf{r}) + \varepsilon(1+r_0^2\nabla^2)^2 x(\mathbf{r}) \quad (2.8)$$

Такий самоузгоджений підхід для потенціалу взаємодії широко використовувався не тільки при дослідженні процесів формування нанорозмірних структур адсорбату при конденсації з газової фази, але й при дослідженні процесів структуроутворення в системах хемічного типу, при дослідженні процесів формування пірамідальних структур при епітаксіальному рості, та при дослідженні процесів формування нанорозмірних кластерів точкових дефектів в твердих тілах, підданих дії опромінювальних джерел.

Функціонал повної вільної енергії має форму

$$F = \int dr \left[ -\frac{\varepsilon}{2} x^2 + x \ln x + (1-x) \ln(1-x) - \frac{\varepsilon}{2} x (1 + r_0^2 \nabla^2)^2 x \right] \quad (2.9)$$

Отже, повний потік можна записати у наступному вигляді

$$J = -DM(x) \nabla \left[ \frac{\delta F_{loc}}{\delta x} - \varepsilon (1 + r_0^2 \nabla^2)^2 x \right] \quad (2.10)$$

### 3.1 СТАЦІОНАРНІ СТАНИ ОДНОРІДНОЇ СИСТЕМИ ТА ЇХ СТІЙКІСТЬ

В данному розділі перевіряю стійкі однорідні стани системи. Однорідні стани зазначають, що  $f(x) = 0$ .

Перевірили як залежить  $k$  – коефіцієнт вертикальної дифузії до  $x$  – концентрації адсорбату при різних значеннях  $\gamma$  (гамма).

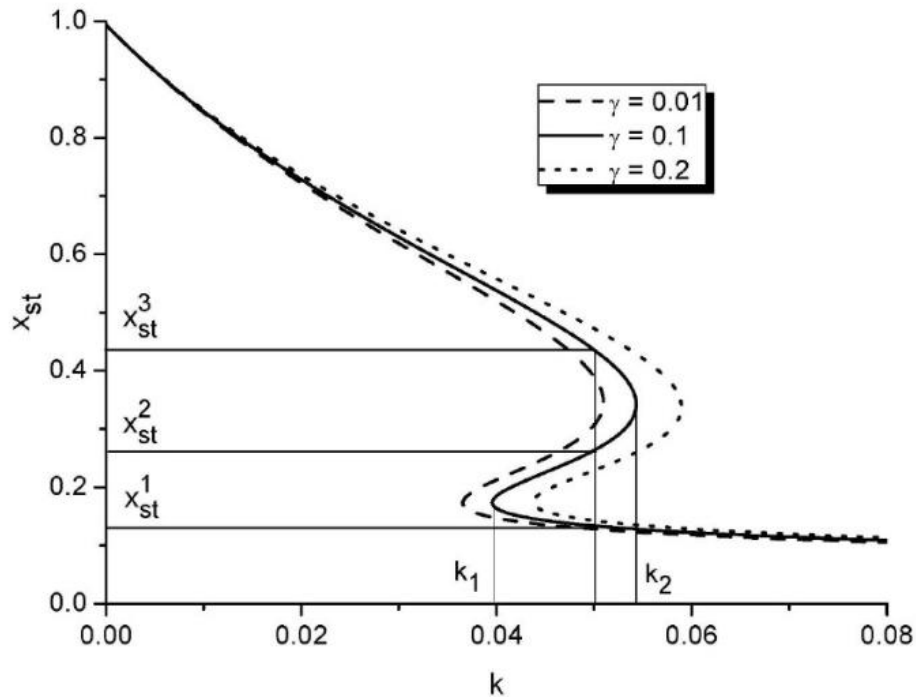


Рисунок 1. Залежність концентрації адсорбату від коефіцієнта дифузії

З графіка видно, що зі збільшенням  $\gamma$ , область  $k$  (коефіцієнту дифузії) зміщується в бік до більших  $k$ .

На малих  $k$  до певного значення  $k_1$ , та на великих  $k$  після певного значення  $k_2$  маємо тільки один стаціонарний стан. В області між  $k_1$  та  $k_2$  маємо 3 стаціонарні стани.

Щоб дослідити, які саме стаціонарні стани: стійкі чи ні, побудуємо залежність  $\lambda_0(k) = \frac{df}{dx} \Big|_{x=x_{st}}$ . Відповідний рисунок представлено на Рис.2., де показано що, нижче кривої коли  $\lambda$  – від'ємна, тобто менша 0, то маємо стійкий стаціонарний стан. Якщо більше 0 то маємо нестійкий стаціонарний стан. Для кожного значення  $k$  маємо 3 стаціонарні точки.



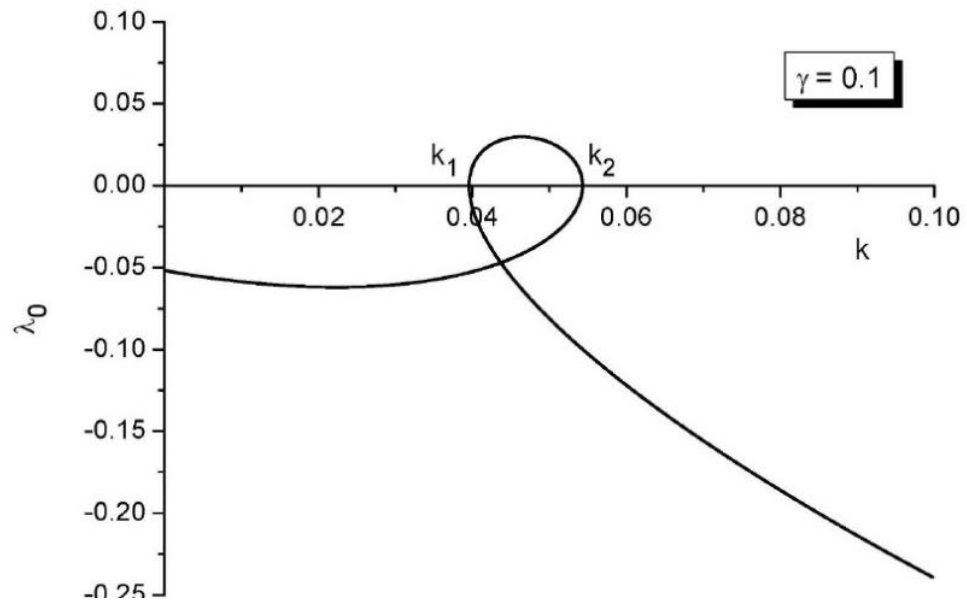


Рисунок 2. Залежність  $\lambda_0$  від  $k$

Тобто за даним графіком показано можливість існування нестійкості системи.

Для того щоб перевірити стійкі чи не стійкі стани системи, будемо діаграму  $k(\gamma)$  і покажемо що між кривими маємо 3 стаціонарні точки.

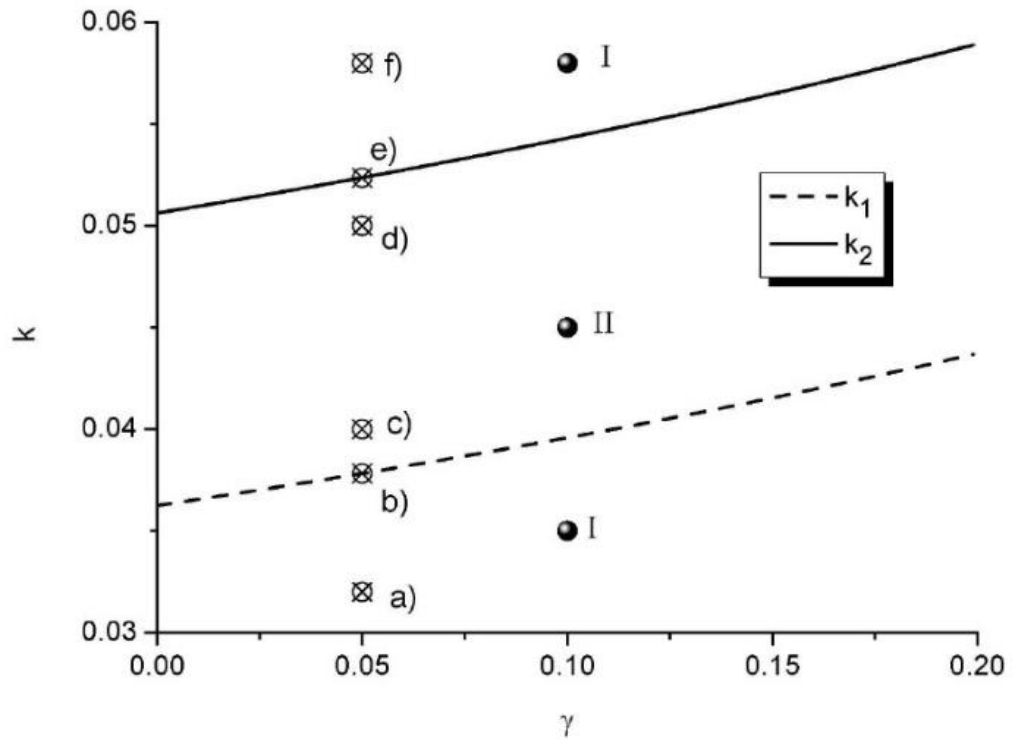


Рисунок 3. Фазова діаграма залежності вертикальної дифузії що залежить від матеріалу

В нашій системі  $\gamma$  приймає різні значення та показуємо як поводить себе область в якій існує фазовий перехід першого роду. (реверсивність)

Щоб переглянути як змінюється потенціал в різних областях фазової діаграми, що відповідає Рис.3, будемо потенціал як:

$$V = - \int f(x) dx, \quad (3.1)$$

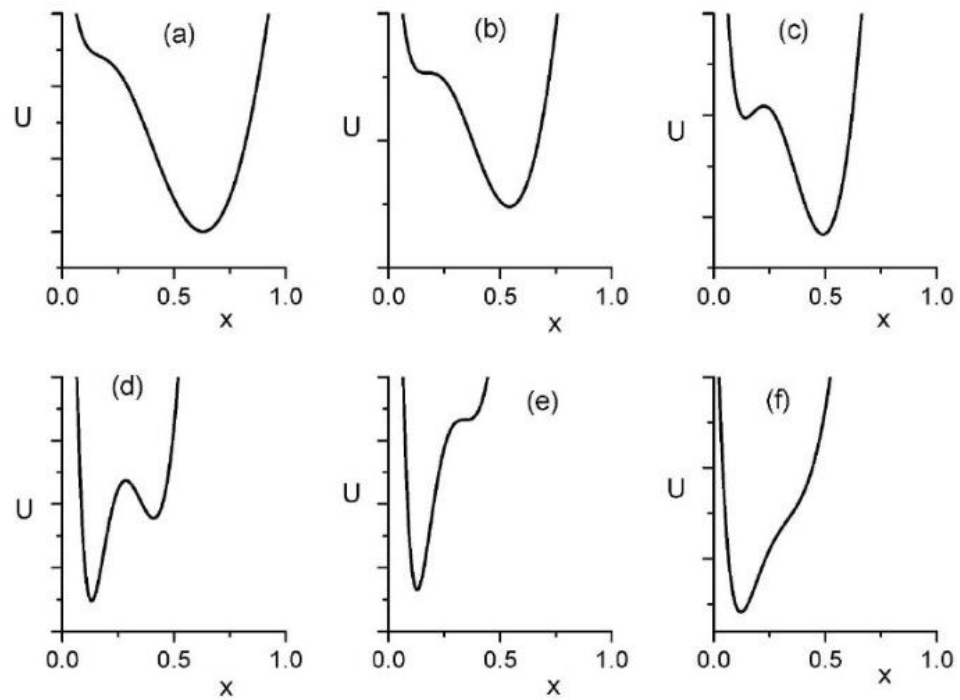


Рисунок 4. Залежність потенціалу в різних областях діаграми рис. 3

Поведінка еволюції системи (тобто керуючого параметру від часу) при різних стаціонарних точках була досліджена з використанням методу Ейлера та методу Рунге-Куты, результати якої представлено на Рис.5 (для одного стаціонарного стану) та Рис.6(для 3 стаціонарних станів).

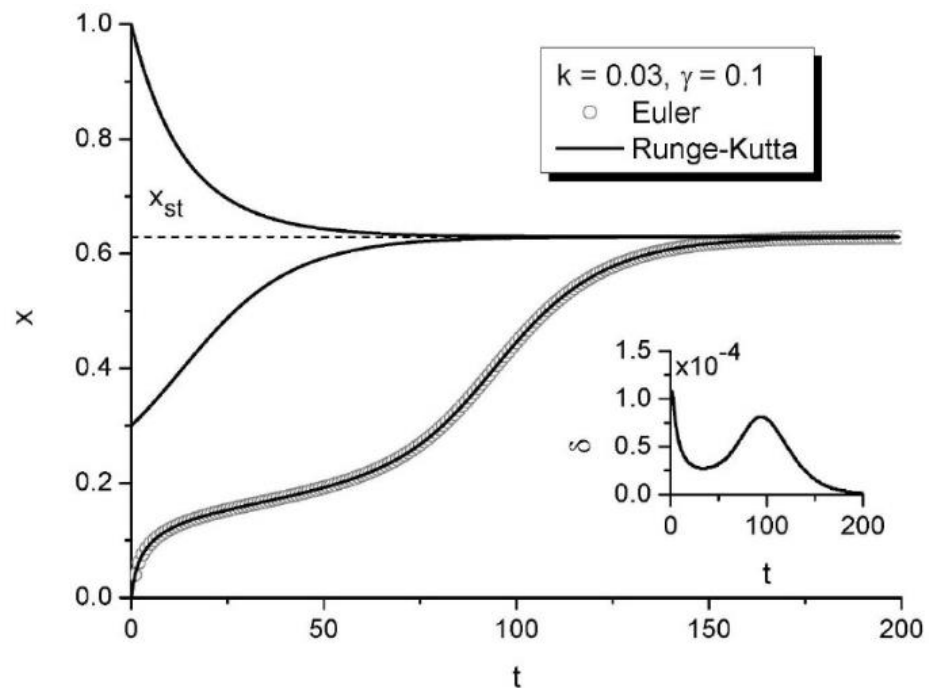


Рисунок 5. Поведінка еволюції системи при різних стаціонарних точках (одна точка)

З рисунків видно, що у випадку стійкого стаціонарного стану траєкторії наближаються до нього, але ніколи його не перейдуть.

Після порахування похибки (вставка на Рис.5) було показано, що похибка (модуль першого від другого) наближається до 0 та вона менша ніж крок інтегрування. Це означає, що можна використати дл вирішення як метод Ейлера, так і метод Рунге-Кута.

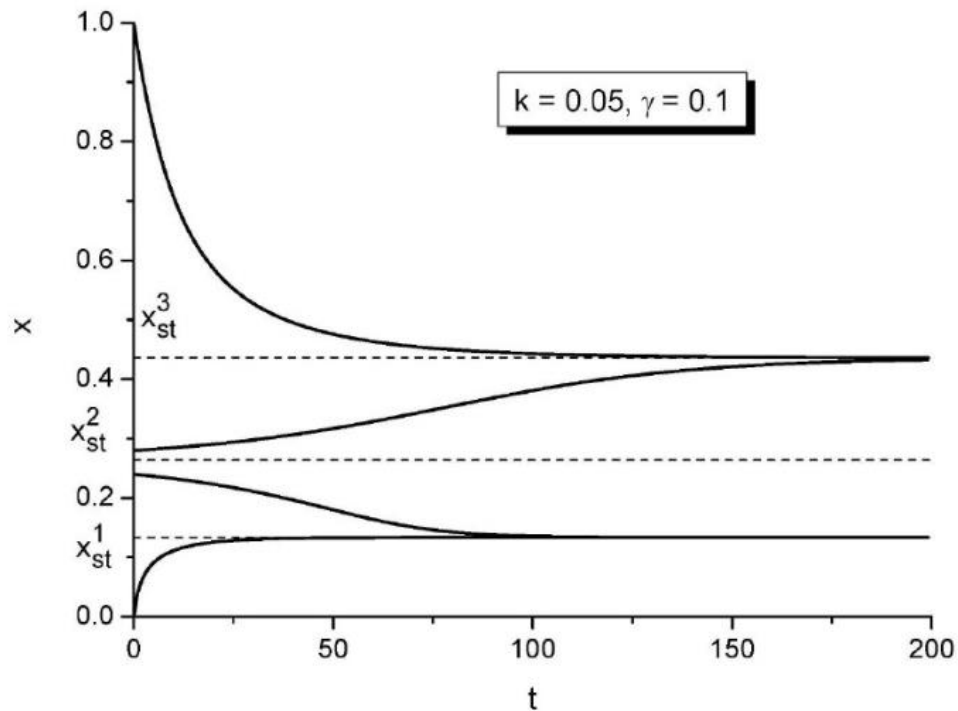


Рисунок 6. Поведінка еволюції системи при різних стаціонарних точках (три точки)

Було показано, що метод Ейлера не дає великої різниці від методу Рунге-Кута. На графіках показано, що траєкторія приближається до стаціонарної точки в стійкому стані та віддаляється від неї в нестійкому.

#### 4.1 УМОВИ РЕАЛІЗАЦІЇ СТІЙКИХ ПРОСТОРОВИХ СТРУКТУР

Розглядаючи відхил концентрації адсорбату від стаціонарного значення  $x_0$ , що відповідає однорідному стійкому станові, усереднені флюктуації у лінійній аналізі можуть розглядатися, як

$$\nabla^2 J = \nabla^2 x - \varepsilon \nabla M(x) \nabla x - \varepsilon \nabla M(x) \nabla L_{sx} x, \quad (4.1)$$

де оператор Свіфта-Хогенбера:

$$L_{sx} x = (1 + \rho_0^2 \nabla^2)^2 x \quad (4.2)$$

Згідно зі стандартною теорією маємо, що у випадку  $\lambda(k) < 0$  всі відхилення від однорідного стану з часом експоненційно затухають і адсорбат однорідно заповнює підкладку.

$$\nabla L_{sx} x = \nabla (1 + \rho_0^2 \nabla^2)^2 x = \nabla x + 2\rho_0^2 \nabla^3 x \quad (4.3)$$

$$\nabla^2 J = \nabla^2 x - \varepsilon \nabla M(x) \nabla x - \varepsilon \nabla M(x) (\nabla x + 2\rho_0^2 \nabla^3 x) \quad (4.4)$$

Переходимо до сферичних координат:

$$x = e^{\lambda t - ikr}, \quad (4.5)$$

інтегруємо по часу:

$$\frac{dx}{dt} = \lambda x, \quad (4.6)$$

інтегруємо по простору:

$$\frac{dx}{dr} = ikx, \quad (4.7)$$

інтегруємо в квадраті:

$$\frac{dx^2}{dr^2} = -k^2 x \quad (4.8)$$

Розкладаємо в ряд Тейлора:

$$\nabla^2 J = -k^2 x + 2k^2 \varepsilon M(x) x - 2\varepsilon M(x) \rho_0^2 k^4 x \quad (4.9)$$

Тут часова змінна  $t$  вимірюється в одиницях. Відповідний показник Ляпунова  $\lambda$  визначається як:

$$\lambda(k) = \lambda_0 - k^2 \{1 - 2\varepsilon M(x_0) [1 - \rho_0^2 k^2]\}, \quad (4.10)$$

де:

$$\lambda_0 = \left. \frac{df}{dx} \right|_{x = x_{st}} \quad (4.11)$$

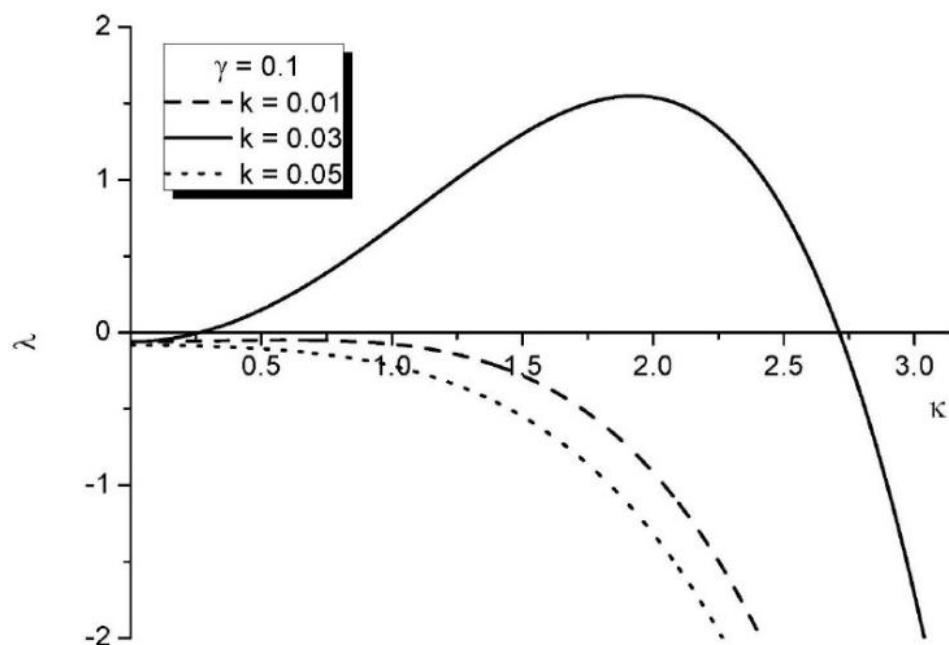


Рисунок 7. Залежність показника Ляпунова

З діаграми стійкості, що представлено на Рис.7 видно, що в області  $\lambda(k) > 0$  (де є нестійкість в системі) будуть реалізуватися стійки структури адсорбату.

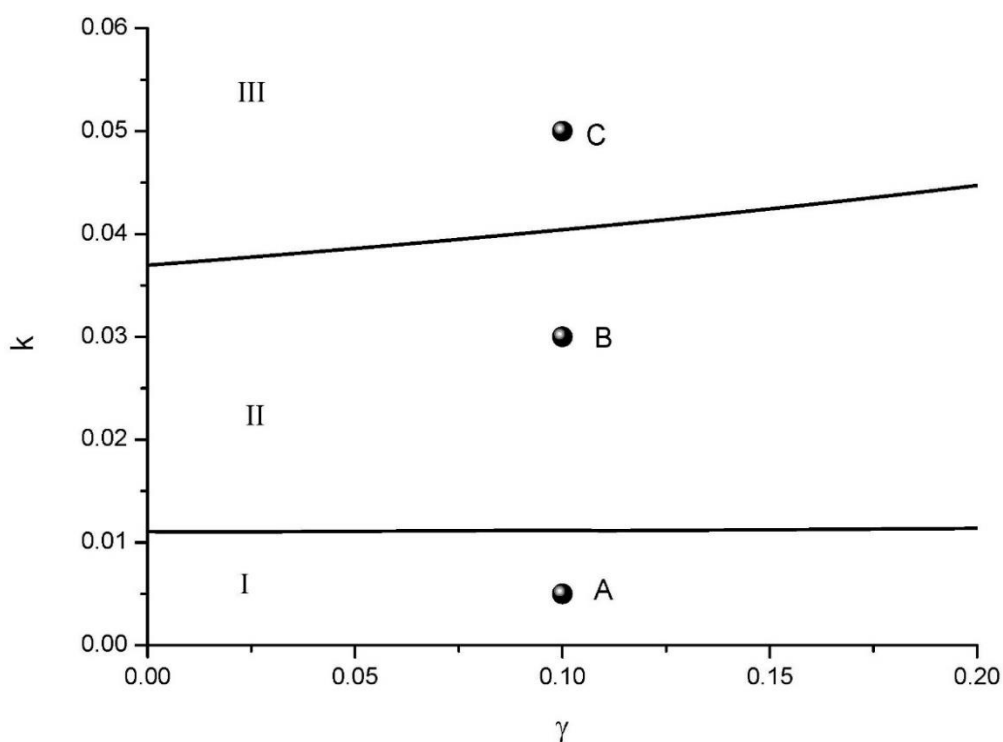


Рисунок 8. Діаграма залежності Рис. 7

Знову показано за допомогою графіка, що в області В будуть сформовані структури. За кордоном області В структури не будуть сформовані.

## 5.1 МОДЕЛЮВАННЯ РОСТУ ПОВЕРХНЕВИХ СТРУКТУР

Досліджуємо зміну морфології поверхні та динаміку утворення поверхневих структур шляхом числових симуляцій. Для цього будемо чисельно розв'язувати рівняння на двовимірній квадратній ґратниці розміром 128x128. Просторові похідні бралися в зворотному просторі з використанням перетворень Фур'є.

Коригуючи і вибираючи малі значення, ми отримуємо структуру з окремими острівцями адсорбата, що сильно розрізняються за лінійними розмірами. Зменшення взаємодії адсорбата призводить до зменшення кількості острівців адсорбата. При збільшенні значень коефіцієнта адсорбції утворюються (ізолювані) вільні острівці.

Такі острівці мають ту ж структуру, що і острівці адсорбата з низькою швидкістю адсорбції.

Моделювання проводилося з кроком інтегрування за часом 0,001 і з кроком інтегрування по решітці 1. В якості початкових умов вибиралося гаусовий розподіл щодо однорідного стаціонарного стану з дисперсією 0,01. Граничні умови на решітці були періодичні.

Основні параметри моделювання:

$$\alpha = 0,06$$

$$\varepsilon = 4,0$$

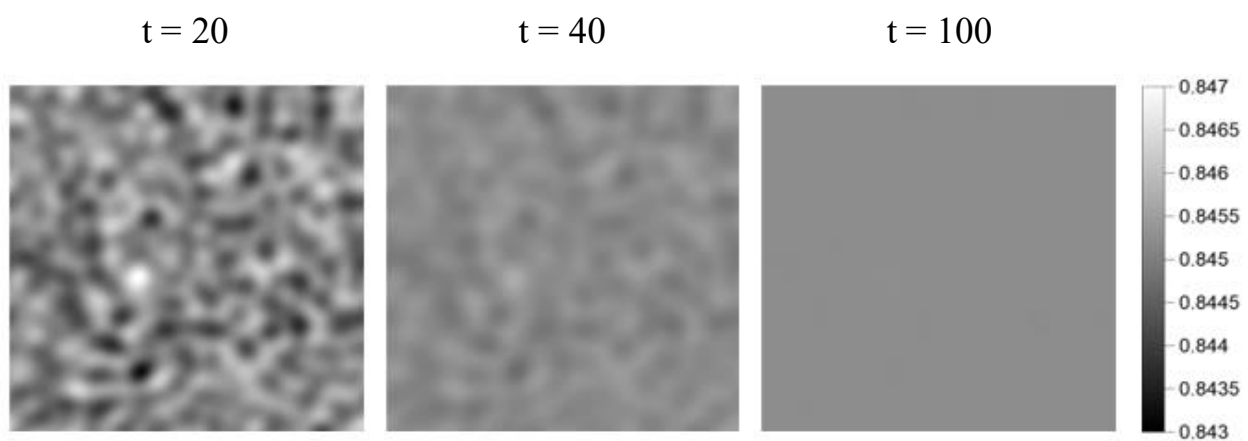
$$D_0 = 1.0$$

$$\rho_0 = 0.5$$

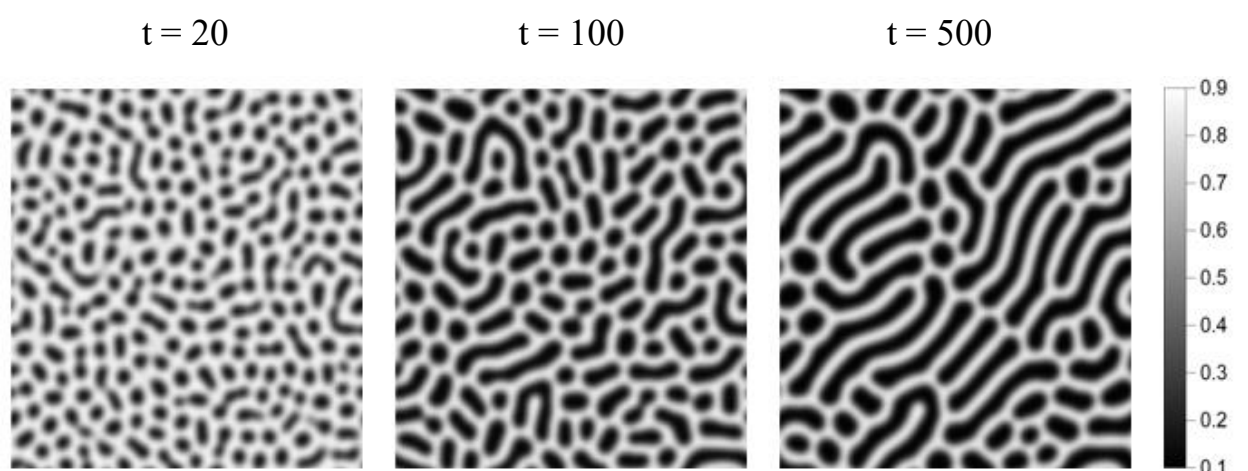
$$\gamma = 0.1$$

$k = 0.005$  (точка А на малюнку 9),  $0.03$  (точка В на малюнку 9),  $0.05$  (точка С на малюнку 9)

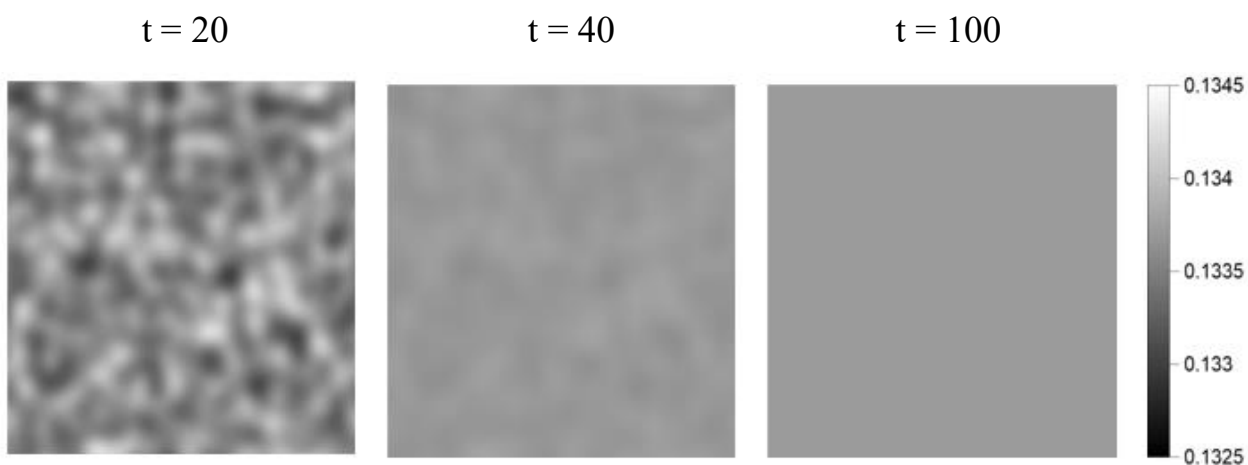
На малюнках справа вказані значення  $x$  в відтінках сірого кольору.



a)



б)



в)

**РИСУНОК 9. ПРОСТОРОВІ СТРУКТУРИ ЩО ВІДПОВІДАЮТЬ МОДЕЛЮВАННЮ РІВНЯННЯ 1.1:** а) Точка А при  $\gamma=0.1$ ,  $k=0.005$ ; б) Точка В при  $\gamma=0.1$ ,  $k=0.03$ ; в) Точка С при  $\gamma=0.1$ ,  $k=0.05$



В областях I і III на малюнку 8 структури є. Тут будь-яке просторове обурення однорідного стаціонарного стану буде затухаючи згодом.

В області I маємо для стаціонарного значення  $x > 0.5$

В області III маємо для стаціонарного значення  $x < 0.5$

В області II будуть просторові структури.

Точки A, B, C показують значення  $\gamma$  і  $k$  при яких проводилося моделювання.

Значення концентрації адатомів подано у відтінках сірого кольору: білі області відповідають наявності адатомів, темні області — без адатомів.

Ситуація критично змінюється з урахуванням незбалансованих хімічних реакцій. Виходячи з цього, такі реакції заморожують структури, які відбуваються на ранніх стадіях, і наступні структури розвиваються дуже повільно.

У результаті конкуренції між хімічними реакціями і процесом взаємодії атомів, такі конструкції стабільні в часі і можуть вважатися стаціонарними.

## ВИСНОВОК

Встановлено, що збільшення коефіцієнту дифузії адатомів між шарами в однорідній системі приводить до реалізації переходів першого роду з формуванням трьох стаціонарних станів.

Виявлено, що збільшення концентрації адсорбату на другому шарі приводить до збільшення значень коефіцієнту дифузії адатомів між шарами коли реалізується бістабільність в системі.

Показано, що просторові структури адсорбату на першому шарі від підкладки будуть формуватися лише за умови врахування переходів адатомів між шарами. Збільшення коефіцієнту вертикальної дифузії приводить до реверсивних процесів формування просторових структур.

## СПИСОК ЛІТЕРАТУРИ

- [1] В. О. Харченко, Д. О. Харченко, С. В. Кохан, В. В. Яновський\* , А. В. Дворниченко: Формування нанорозмірних структур адсорбату у процесах конденсації з газової фази.
- [2] Наноматеріали та структури на їх основі, оптичні та квантові комп'ютери. Євтух А. А., д. ф.-м. н., професор, 2013р.
- [3] Vasyl O Kharchenko<sup>1,2</sup> , Dmitrii O Kharchenko<sup>1</sup> , Sergei V Kokhan<sup>1</sup> , Iryna V Vernyhora<sup>1</sup> and Vladimir V Yanovsky: Properties of nano-islands formation in nonequilibrium reaction–diffusion systems with memory effects.
- [4] М.Г. Рамбиди: Нанотехнологии и молекулярные компьютеры. – М.:ФИЗМАТЛИТ, 2007г.
- [5] Лапшин Владимир Ильич, Волков Андрей Николаевич, Кульков Анатолий Николаевич, Константинов Александр Андреевич. К вопросу классификации пластовых флюидов нефтегазоконденсатных залежей.
- [6] Кельцев Н. В. Основы адсорбционной техники. - М.: 1984. - 592 с.
- [7] Карнаухов А.П. Адсорбция. Текстура дисперсных и пористых материалов. — Новосибирск: Наука. 1999. — 470 с.
- [8] Мала гірнича енциклопедія : у 3 т. / за ред. В. С. Білецького. — Д. : Східний видавничий дім, 2004—2013.
- [9] Кундеева, Г. О. Сутність горизонтальної та вертикальної дифузії інноваційного процесу / Г. О. Кундеева // Сучасні проблеми інноваційного розвитку держави : матеріали III Міжнар. наук.-практ. конф. – Д. : ПДАБА, 2008. – Т. 4. – С. 62–66.
- [10] Глосарій термінів з хімії // Й.Опейда, О.Швайка. Інститут фізико-органічної хімії та вуглехімії ім. Л. М. Литвиненка НАН України, Донецький національний університет — Донецьк: «Вебер», 2008. — 758 с.

- [11] Елизаров Д. П., Елькин А. И., Даванков В. А. и др. Экспериментальное изучение сорбционной активности распространенных адсорбентов // Эфферентная терапия,— 2003.— Т. 9
- [12] Константи рівноваги квазіхімічних реакцій дефектоутворення в CdTe / В.В. Прокопів, І.В. Горічок, У.М. Писклинець // Український хімічний журнал. — 2009. — Т. 75
- [13] Изучение процесса периодической адсорбции: методические указания к лабораторной работе / сост.: Д.В. Башкиров, А.И. Разинов, А.В. Клинов, И.П. Анашкин — Казань: изд-во КНИТУ, 2013 – 21 с
- [14] Дытнерский, Ю.И. Процессы и аппараты химической технологии: в 2 кн. / Ю.И.Дытнерский. – 3-е изд. – М.: Химия, 2002. – 768 с

## ДОДАТОК А

```
#include<math.h>
#include<string.h>
#include<stdio.h>
#include<stdlib.h>
double alpha,epsilon,kv,gamma;
double y(double x)
{
    double z=gamma*x;
    return(z);
}
double dy(double x)
{
    double z=gamma;
    return(z);
}
double fa(double x)
{
    double z=(1.0-x)*(1.0-y(x));
    return(z);
}
double fd(double x)
{
    double z=-x*(1.0-y(x))*exp(-2.0*epsilon*x);
    return(z);
}
double fv(double x)
{
    double z=y(x)-x;
    return(z);
}
```

```
}  
double k_st(double x)  
{  
    double z=-(\alpha*fa(x)+fd(x))/fv(x);  
    return(z);  
}  
int main(int argc, char *argv[])  
{  
    double x;  
    double dx=0.001;  
    double quan;  
    epsilon=4.0;  
    alpha=0.06;  
    kv=0.01;  
    gamma=0.1;  
    char FNAME[40]="x(k)a006e4g01.dat";  
    FILE *write_file=fopen(FNAME,"w+");  
    fclose(write_file);  
    for(x=2.0*dx;x<=1.0+dx;x+=0.001)  
    {  
        quan=k_st(x);  
        //if ((quan>1.0)||(quan<0.0)) break;  
        if(quan>0.0)  
        {  
            write_file=fopen(FNAME,"a+");  
            fprintf(write_file,"%lf\t%lf\n",quan,x);  
            printf("%lf\t%lf\n",quan,x);  
            fclose(write_file);  
        }  
    }  
}
```

```
    fclose(write_file);
    return(0);
}

#include<math.h>
#include<string.h>
#include<stdio.h>
#include<stdlib.h>
double alpha,epsilon,kv,gamma;
double y(double x)
{
    double z=gamma*x;
    return(z);
}
double dy(double x)
{
    double z=gamma;
    return(z);
}
double fa(double x)
{
    double z=(1.0-x)*(1.0-y(x));
    return(z);
}
double fd(double x)
{
    double z=-x*(1.0-y(x))*exp(-2.0*epsilon*x);
    return(z);
}
```

```

double fv(double x)
{
    double z=y(x)-x;
    return(z);
}
double k_st(double x)
{
    double z=-(alpha*fa(x)+fd(x))/fv(x);
    return(z);
}
double dfdx(double x)
{
    double dfa=-alpha*(1.0-y(x)+(1.0-x)*(dy(x)));
    double dfd=exp(-2.0*epsilon*x)*((2.0*epsilon*x-1.0)*(1.0-y(x))+x*dy(x));
    double dfv(kv*(dy(x)-1.0));
    double z=dfa+dfd+dfv;
    return(z);
}
int main(int argc, char *argv[])
{
    double x;
    double dx=0.001;
    double quan;
    double lyam0;
    epsilon=4.0;
    alpha=0.06;
    kv=0.01;
    gamma=0.1;
    char FNAME[40]="I0(k)a006e4g01.dat";

```



```

FILE *write_file=fopen(FNAME,"w+");
fclose(write_file);
for(x=2.0*dx;x<=1.0+dx;x+=0.001)
{
    quan=k_st(x);
    //if ((quan>1.0)||((quan<0.0)) break;
    if((quan>0.0)&&(quan<0.1))
    {
        kv = quan;
        lyam0=dfdx(x);
        write_file=fopen(FNAME,"a+");
        fprintf(write_file,"%lf\t%lf\t%lf\n",quan,x,lyam0);
        printf("%lf\t%lf\n",quan,x);
        fclose(write_file);
    }
}
fclose(write_file);
return(0);
}

```

```

#include<math.h>
#include<string.h>
#include<stdio.h>
#include<stdlib.h>
double alpha,epsilon,kv,gamma;
double y(double x)
{
    double z=gamma*x;
    return(z);
}

```

```

double dy(double x)
{
    double z=gamma;
    return(z);
}
double fa(double x)
{
    double z=(1.0-x)*(1.0-y(x));
    return(z);
}
double fd(double x)
{
    double z=-x*(1.0-y(x))*exp(-2.0*epsilon*x);
    return(z);
}
double fv(double x)
{
    double z=y(x)-x;
    return(z);
}
double k_st(double x)
{
    double z=-(alpha*fa(x)+fd(x))/fv(x);
    return(z);
}
double dfdx(double x)
{
    double dfa=-alpha*(1.0-y(x)+(1.0-x)*(dy(x)));
    double dfd=exp(-2.0*epsilon*x)*((2.0*epsilon*x-1.0)*(1.0-y(x))+x*dy(x));
    double dfv=kv*(dy(x)-1.0);

```

```

double z=dfa+dfd+dfv;
return(z);
}
int main(int argc, char *argv[])
{
double x;
double dx=0.001;
double quan;
double lyam0;
epsilon=4.0;
alpha=0.06;
kv=0.01;
gamma=0.1;
double k_1,k_2,k_3;
char FNAME1[40]="k(g)e4a006_max.dat";
FILE *write_file1=fopen(FNAME1,"w+");
fclose(write_file1);
for(gamma=0.0;gamma<=1.0;gamma+=0.001)
for(x=2.0*dx;x<1.0;x+=0.001)
{
k_1=k_st(x-dx);
k_2=k_st(x);
k_3=k_st(x+dx);
if((k_2>k_1)&&(k_2>k_3)&&(k_2>0.0)) //maximum
{
write_file1=fopen(FNAME1,"a+");
fprintf(write_file1,"%lf\t%lf\n",gamma,k_2,x);
printf("%lf\t%lf\n",gamma,k_2);
fclose(write_file1);
break;
}
}
}

```

```

    }
}
char FNAME2[40]="k(g)e4a006_min.dat";
FILE *write_file2=fopen(FNAME2,"w+");
fclose(write_file2);
for(gamma=0.0;gamma<=1.0;gamma+=0.001)
for(x=2.0*dx;x<1.0;x+=0.001)
{
    k_1=k_st(x-dx);
    k_2=k_st(x);
    k_3=k_st(x+dx);
    if((k_2<k_1)&&(k_2<k_3)&&(k_2>0.0)) //minimum
    {
        write_file2=fopen(FNAME2,"a+");
        fprintf(write_file2,"%lf\t%lf\n",gamma,k_2,x);
        printf("%lf\t%lf\n",gamma,k_2);
        fclose(write_file2);
        break;
    }
}
return(0);
}

```

```

#include<math.h>
#include<string.h>
#include<stdio.h>
#include<stdlib.h>
double alpha,epsilon,kv,gamma;
double y(double x)
{

```

```
double z=gamma*x;
return(z);
}
double dy(double x)
{
double z=gamma;
return(z);
}
double fa(double x)
{
double z=(1.0-x)*(1.0-y(x));
return(z);
}
double fd(double x)
{
double z=-x*(1.0-y(x))*exp(-2.0*epsilon*x);
return(z);
}
double fv(double x)
{
double z=y(x)-x;
return(z);
}
double f(double x)
{
double z=alpha*fa(x)+fd(x)+kv*fv(x);
return(z);
}
int main(int argc, char *argv[])
{
```

```
double x;
epsilon=4.0;
alpha=0.06;
kv=0.01;
gamma=0.1;
char FNAME[20]="pot3.dat";
FILE *write_file=fopen(FNAME,"w+");
double U1,scale=100.0;
dx=0.001;
for(x=4.0*dx;x<1.0;x+=dx)
{
    U1+=-0.5*(f(x-dx)+f(x))*dx*scale;
    fprintf(write_file,"%lf\t%lf\n",x,U1);
}
fclose(write_file);
return(0);
}
```

## ДОДАТОК В

```
#include<math.h>
#include<string.h>
#include<stdio.h>
#include<stdlib.h>
double alpha,epsilon,kv,gamma;
double y(double x)
{
    double z=gamma*x;
    return(z);
}
double dy(double x)
{
    double z=gamma;
    return(z);
}
double fa(double x)
{
    double z=(1.0-x)*(1.0-y(x));
    return(z);
}
double fd(double x)
{
    double z=-x*(1.0-y(x))*exp(-2.0*epsilon*x);
    return(z);
}
double fv(double x)
{
    double z=y(x)-x;
    return(z);
}
```

```
}  
double f(double x)  
{  
    double z=alpha*fa(x)+fd(x)+kv*fv(x);  
    return(z);  
}  
double Euler(double x)  
{  
    double z = x + dt*f(x);  
    return(z);  
}  
double RK4(double x)  
{  
    double k1, k2, k3, k4;  
    k1 = f(x);  
    k2 = f(x+dt*k1/2.0);  
    k3 = f(x+dt*k2/2.0);  
    k4 = f(x+dt*k3);  
    double z = x + dt*(k1 + 2.0*k2 + 2.0*k3 + k4)/6.0;  
    return(z);  
}  
int main(int argc, char *argv[])  
{  
    double x;  
    FILE *write_file=fopen(FNAME,"w+");  
    fclose(write_file);  
    dt = 0.01;  
    double tfin = 200.0;  
    double t = 0;  
    double tprint = 1.0;
```



```

double dtprint = 1.0;
epsilon= 4.0;
alpha = 0.06;
gamma = 0.1;
kv = 0.03;
x=0.0;
double y = x;
write_file=fopen(FNAME,"a+");
fprintf(write_file,"%lf\t%lf\t%lf\t%lf\n",t,x,y,fabs(x-y));
fclose(write_file);
while(t<= tfin)
{
    x = Euler(x); // euler
    y = RK4(y); //RK
    if(t>=tprint)
    {
        write_file=fopen(FNAME,"a+");
        fprintf(write_file,"%lf\t%lf\t%lf\t%lf\n",t,x,y,fabs(x-y));
        printf("%lf\t%lf\n",t,x);
        fclose(write_file);
        tprint+=dtprint;
    }
    t+=dt;
}
fclose(write_file);
return(0);
}

#include<math.h>
#include<string.h>

```

```
#include<stdio.h>
#include<stdlib.h>
#define Pi 4.0*atan(1.0)
double alpha,epsilon,kv,gamma;
double y(double x)
{
    double z=gamma*x;
    return(z);
}
double dy(double x)
{
    double z=gamma;
    return(z);
}
double mu(double x)
{
    double z=x*(1.0-x);
    return(z);
}
double fa(double x)
{
    double z=(1.0-x)*(1.0-y(x));
    return(z);
}
double fd(double x)
{
    double z=-x*(1.0-y(x))*exp(-2.0*epsilon*x);
    return(z);
}
double fv(double x)
```

```

{
double z=y(x)-x;
return(z);
}
double f(double x)
{
double z=alpha*fa(x)+fd(x)+kv*fv(x);
return(z);
}
double x_hom()
{
double x,dx=0.001;
for(x=2.0*dx;x<1.0;x+=dx)
{
if(((f(x-dx)<0.0)&&(f(x+dx)>0.0))||((f(x-dx)>0.0)&&(f(x+dx)<0.0))) break;
}
return(x);
}
double dfdx(double x)
{
double dfa=-alpha*(1.0-y(x)+(1.0-x)*(dy(x)));
double dfd=exp(-2.0*epsilon*x)*((2.0*epsilon*x-1.0)*(1.0-y(x))+x*dy(x));
double dfv=kv*(dy(x)-1.0);
double z=dfa+dfd+dfv;
return(z);
}
double lyamk(double x, double k)
{
double y=-k*k*(1.0-2.0*epsilon*mu(x)*(1.0-k*k*rho*rho));
return(y);
}

```

```
}  
int main(int argc, char *argv[])  
{  
    double x;  
    double dx=0.001;  
    double k, dk=0.001;  
    double lyam_curr;  
    double rho=0.25;  
    double lyam0;  
    kv=0.01;  
    gamma=0.1;  
    char FNAME[40]="L(k)a006e4kv001g01.dat";  
    FILE *write_file=fopen(FNAME,"w+");  
    fclose(write_file);  
    x=x_hom();  
    lyam0=dfdx(x);  
    if(lyam0<0.0)  
    {  
        for(k=dk;k<Pi+2.0*dk;k+=dk)  
        {  
            lyam_curr=lyam0+lyamk(x,k);  
            write_file=fopen(FNAME,"a+");  
            fprintf(write_file,"%lf\t%lf\n",k,lyam_curr);  
            fclose(write_file);  
        }  
    }  
    fclose(write_file);  
    return(0);  
}
```

```
#include<math.h>
#include<string.h>
#include<stdio.h>
#include<stdlib.h>
#define Pi 4.0*atan(1.0)
double alpha,epsilon,kv,gamma;
double y(double x)
{
    double z=gamma*x;
    return(z);
}
double dy(double x)
{
    double z=gamma;
    return(z);
}
double mu(double x)
{
    double z=x*(1.0-x);
    return(z);
}
double fa(double x)
{
    double z=(1.0-x)*(1.0-y(x));
    return(z);
}
double fd(double x)
{
    double z=-x*(1.0-y(x))*exp(-2.0*epsilon*x);
    return(z);
}
```

```

}
double fv(double x)
{
double z=y(x)-x;
return(z);
}
double f(double x)
{
double z=alpha*fa(x)+fd(x)+kv*fv(x);
return(z);
}
double x_hom()
{
double x,dx=0.001;
for(x=2.0*dx;x<1.0;x+=dx)
{
if(((f(x-dx)<0.0)&&(f(x+dx)>0.0))||((f(x-dx)>0.0)&&(f(x+dx)<0.0))) break;
}
return(x);
}
double dfdx(double x)
{
double dfa=-alpha*(1.0-y(x)+(1.0-x)*(dy(x)));
double dfd=exp(-2.0*epsilon*x)*((2.0*epsilon*x-1.0)*(1.0-y(x))+x*dy(x));
double dfv=kv*(dy(x)-1.0);
double z=dfa+dfd+dfv;
return(z);
}
double lyamk(double x, double k)
{

```

```

double y=-k*k*(1.0-2.0*epsilon*mu(x)*(1.0-k*k*rho*rho));
return(y);
}
int main(int argc, char *argv[])
{
double x;
double dx=0.001;
double k, dk=0.001;
double lyam_curr;
double rho=0.25;
double lyam0;
kv=0.01;
gamma=0.1;
int flag = 0;
char FNAME[40]="k(g)a006e45inh_1.dat";
FILE *write_file=fopen(FNAME,"w+");
fclose(write_file);
//for(kv=0.0;kv<=0.1;kv+=0.001)
for(gamma=0.0;gamma<=1.0;gamma+=0.01)
//for(kv=0.1;kv>0.0;kv-=0.001)
{
printf("%lf\n",gamma);
//for(kv=1.0;kv>0.0;kv-=0.001)
for(kv=0.0;kv<=1.0;kv+=0.001)
//for(gamma=0.0;gamma<0.1;gamma+=0.001)
{
x=x_hom();
if((x>0.0)&(x<=1.0))
{
lyam0=dfdx(x);

```

```
if(lyam0<0.0)
{
for(k=dk;k<Pi+2.0*dk;k+=dk)
{
lyam_prev=lyam0+lyamk(x,k-dk);
lyam_curr=lyam0+lyamk(x,k);
if((lyam_prev<0.0)&&(lyam_curr>0.0))
{
write_file=fopen(FNAME,"a+");
fprintf(write_file,"%lf\t%lf\n",gamma,kv);
fclose(write_file);
flag=1;
break;
}
}
}
if(flag==1) { flag=0;break;}
}
}
fclose(write_file);
return(0);
}
```



## ДОДАТОК С

```
#include <cmath>
#include <fftw3.h>
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/stat.h>
#define Nx 128
#define Ny 128
#define dx 1.0
#define dy 1.0
#define alpha 0.06
#define eps 4.0

#define rho0 0.5
#define D0 1.0
#define dt 0.001
#define tini 0.0
#define tmax 500.0
#define dtprint 20.0

double *y = new double[Nx*Ny];
double *ny = new double[Nx*Ny];
double kv, gama;
double *k2 = new double[(Nx+2)*(Ny+2)];
char fname[50];
char dname[80];
```

```
double x2(double x)
{
    double z=gama*x;
    return(z);
}
double fa(double x)
{
    double z=(1.0-x)*(1.0-x2(x));
    return(z);
}
double fd(double x)
{
    double z=-x*(1.0-x2(x))*exp(-2.0*eps*x);
    return(z);
}
double fv(double x)
{
    double z=x2(x)-x;
    return(z);
}
double F(double x)
{
    double z=alpha*fa(x)+fd(x)+kv*fv(x);
    return(z);
}
double M(double x)
{
    double res = x*(1.0-x);
    return(res);
}
```

```

double G(double x)
{
    double res = x;
    return(res);
}
double x_hom()
{
    double x, dxx=0.001;
    for(x=2.0*dxx;x<1.0;x+=dxx)
    {
        if(((F(x-dxx)<0.0)&&(F(x+dxx)>0.0))||((F(x-dxx)>0.0)&&(F(x+dxx)<0.0)))
        break;
    }
    return(x);
}
double RandG(double Mean, double StdDev)
{
    double U1, S2;
    while (1)
    {
        U1 = 2.0*rand()/(RAND_MAX+1.0) - 1.0;
        S2 = U1*U1 + pow((2.0*rand()/(RAND_MAX+1.0)-1.0),2.0);
        if (S2<1.0) break;
    }
    double rez=sqrt(-2.0*log(S2)/S2)*U1*StdDev+Mean;
    return(rez);
}
void initialData()
{
    gama = 0.1:

```

```

kv = 0.03;
for (int i = 0; i<Nx; i++)
    for (int j = 0; j<Ny; j++)
        y[j+Ny*i] = RandG(x_hom(), 0.01);
int Nx21 = Nx / 2 + 1;
int Ny21 = Ny / 2 + 1;
int Nx2 = Nx + 2;
int Ny2 = Ny + 2;
double *kx = new double[Nx+2];
double *ky = new double[Ny+2];
double fk1,fk2;
double delkx = (2.0*M_PI)/(Nx*dx);
double delky = (2.0*M_PI)/(Ny*dy);
for(int i = 0; i < Nx21; i++)
{
    fk1 = i * delkx;
    kx[i] = fk1;
    kx[Nx2 - i] = -fk1;
}
for(int j = 0; j < Ny21; j++)
{
    fk2 = j * delky;
    ky[j] = fk2;
    ky[Ny2 - j] = -fk2;
}
for (int i = 0; i<Nx; i++)
    for (int j = 0; j<Ny; j++)
        k2[j+Ny*i] = kx[i]*kx[i] + ky[j]*ky[j];
free(kx);
free(ky);

```

```

}
void FFT(double *inp, double *out)
{
    fftw_complex *inp_x, *inp_G, *out_x, *out_G, *out_dx_k, *out_dx_r;
    inp_x = (fftw_complex*)fftw_malloc(sizeof(fftw_complex)*(Nx*Ny));
    inp_G = (fftw_complex*)fftw_malloc(sizeof(fftw_complex)*(Nx*Ny));
    out_x = (fftw_complex*)fftw_malloc(sizeof(fftw_complex)*(Nx*Ny));
    out_G = (fftw_complex*)fftw_malloc(sizeof(fftw_complex)*(Nx*Ny));
    out_dx_k = (fftw_complex*)fftw_malloc(sizeof(fftw_complex)*(Nx*Ny));
    out_dx_r = (fftw_complex*)fftw_malloc(sizeof(fftw_complex)*(Nx*Ny));
    for (int i = 0; i<Nx*Ny; i++)
    {
        inp_x[i][0] = inp[i];
        inp_x[i][1] = 0.0;
        inp_G[i][0] = G(inp[i]);
        inp_G[i][1] = 0.0;
    }
    fftw_plan plan_x = fftw_plan_dft_2d(Nx, Ny, inp_x, out_x, FFTW_FORWARD,
    FFTW_ESTIMATE);
    fftw_execute(plan_x);
    fftw_plan plan_G = fftw_plan_dft_2d(Nx, Ny, inp_G, out_G,
    FFTW_FORWARD, FFTW_ESTIMATE);
    fftw_execute(plan_G);
    fftw_complex *fnc_r, *fnc_k;
    fnc_r = (fftw_complex*)fftw_malloc(sizeof(fftw_complex)*(Nx*Ny));
    fnc_k = (fftw_complex*)fftw_malloc(sizeof(fftw_complex)*(Nx*Ny));
    for (int i = 0; i<Nx*Ny; i++)
    {
        fnc_k[i][0] = - sqrt(k2[i]) * (out_G[i][1] - rho0*rho0*k2[i]*out_x[i][1]);
        fnc_k[i][1] = sqrt(k2[i]) * (out_G[i][0] - rho0*rho0*k2[i]*out_x[i][0]);
    }
}

```

```

}
fftw_plan plan_k2r = fftw_plan_dft_2d(Nx, Ny, fnc_k, fnc_r,
FFTW_BACKWARD, FFTW_ESTIMATE);
fftw_execute(plan_k2r);
for (int i = 0; i<Nx*Ny; i++)
{
    fnc_r[i][0] = (1.0/Nx/Ny)*fnc_r[i][0] * M(inp[i]);
    fnc_r[i][1] = (1.0/Nx/Ny)*fnc_r[i][1] * M(inp[i]);
}
fftw_plan plan_r2k = fftw_plan_dft_2d(Nx, Ny, fnc_r, fnc_k,
FFTW_FORWARD, FFTW_ESTIMATE);
fftw_execute(plan_r2k);
for (int i = 0; i<Nx*Ny; i++)
{
    out_dx_k[i][0] = - k2[i]*out_x[i][0] + 2.0*eps*sqrt(k2[i])*fnc_k[i][1];
    out_dx_k[i][1] = - k2[i]*out_x[i][1] - 2.0*eps*sqrt(k2[i])*fnc_k[i][0];
}
fftw_plan plan_out = fftw_plan_dft_2d(Nx, Ny, out_dx_k, out_dx_r,
FFTW_BACKWARD, FFTW_ESTIMATE);
fftw_execute(plan_out);
for (int i = 0; i<Nx*Ny; i++)
    out[i] = (1.0/Nx/Ny)*out_dx_r[i][0];
fftw_free(inp_x);
fftw_free(inp_G);
fftw_free(out_x);
fftw_free(out_G);
fftw_free(fnc_r);
fftw_free(fnc_k);
fftw_free(out_dx_r);
fftw_free(out_dx_k);

```

```

fftw_destroy_plan(plan_x);
fftw_destroy_plan(plan_G);
fftw_destroy_plan(plan_k2r);
fftw_destroy_plan(plan_r2k);
fftw_destroy_plan(plan_out);
}
void check_and_set(int i)
{
    if(y[i] < 0.001) y[i] = 0.001;
    if(y[i] > 0.999) y[i] = 0.999;
}
void rwrite_out_files(double t)
{
    sprintf(fname,"k%.3fg%.1fout_%i.txt",kv,gama,int(t));
    FILE *fout = fopen(fname,"w");
    for (int i = 0; i < Nx; i++)
        for (int j = 0; j < Ny; j++)
            fprintf(fout,"%i\t%i\t%f\n",i,j,y[j+Ny*i]);
    fclose(fout);
}
int main()
{
    double t = tini,
           tprint = tini;
    time_t tm;
    time(&tm);
    srand((long int)tm);
    initialData();
    while (t <= tmax)
    {

```

```
FFT(y, ny);
for (int i = 0; i < Nx*Ny; i++)
{
    y[i] += dt * ( F(y[i]) + D0*ny[i] );
    check_and_set(i);
}
if((t>tprint-dt)&&(t<tprint+dt))
{
    rwrite_out_files(tprint);
    tprint += dtprint;
}
t += dt;
}
free(y);
free(ny);
free(k2);
return 0;
}
```